

Conventions, Accuracy Metrics, Classification, Regression

Nipun Batra

July 21, 2025

IIT Gandhinagar

- Complete PoseNet Demo

- [Complete PoseNet Demo](#)
- [Blog post from Google](#)

- [Complete PoseNet Demo](#)
- [Blog post from Google](#)
- [Rock Paper Scissors](#)

Revision: What is Machine Learning

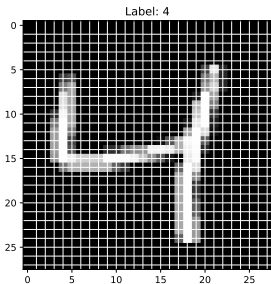
“Field of study that gives computers the ability to learn without being explicitly programmed” - Arthur Samuel [1959]

Revision: What is Machine Learning

“Field of study that gives computers the ability to learn without being explicitly programmed” - Arthur Samuel [1959]

Let us work on the digit recognition problem.

Notebook: rule-based-vs-ml.html



Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: $| + \text{---} + | +$ another vertically down $|$

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: $| + \text{---} + | + \text{another vertically down } |$

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: | + — + | + another vertically down |
- The heights of each of the | need to be similar within tolerance

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: | + — + | + another vertically down |
- The heights of each of the | need to be similar within tolerance

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: | + — + | + another vertically down |
- The heights of each of the | need to be similar within tolerance
- Each of the | can be slightly slanted. Similarly the horizontal line can be slanted.

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: | + — + | + another vertically down |
- The heights of each of the | need to be similar within tolerance
- Each of the | can be slightly slanted. Similarly the horizontal line can be slanted.

Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: | + — + | + another vertically down |
- The heights of each of the | need to be similar within tolerance
- Each of the | can be slightly slanted. Similarly the horizontal line can be slanted.
- There can be some cases of 4 where the first | is at 45 degrees

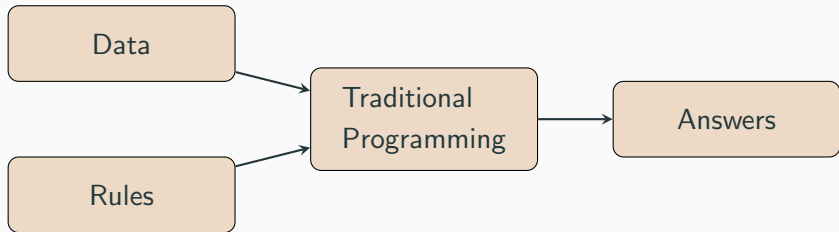
Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: | + — + | + another vertically down |
- The heights of each of the | need to be similar within tolerance
- Each of the | can be slightly slanted. Similarly the horizontal line can be slanted.
- There can be some cases of 4 where the first | is at 45 degrees

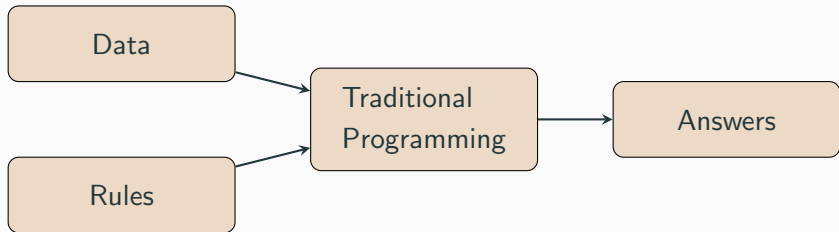
Revision: What is Machine Learning

- How would you program to recognise digits? Start with 4.
- Maybe 4 can be thought of as: | + — + | + another vertically down |
- The heights of each of the | need to be similar within tolerance
- Each of the | can be slightly slanted. Similarly the horizontal line can be slanted.
- There can be some cases of 4 where the first | is at 45 degrees
- There can be some cases of 4 where the width of each stroke is different

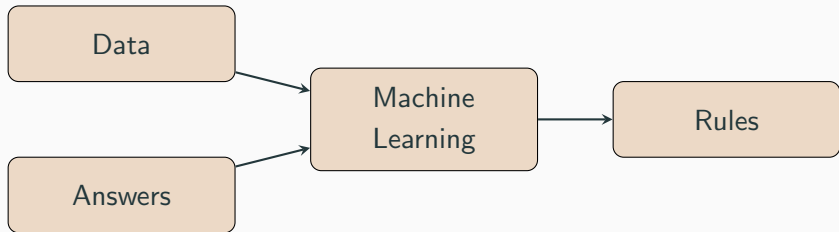
Traditional Programming vs Machine Learning



Traditional Programming



Machine Learning



Revision: What is Machine Learning

“A computer program is said to learn from experience E with respect to some class of tasks T and performance measure P if its performance at tasks in T , as measured by P , improves with experience E .” - Tom Mitchell

First ML Task: Grocery store tomato quality prediction

Problem statement: You want to predict the quality of a tomato given its visual features.

Imagine you have some past data on quality of tomatoes. What visual features do you think will be useful?

Imagine you have some past data on quality of tomatoes. What visual features do you think will be useful?

Imagine you have some past data on quality of tomatoes. What visual features do you think will be useful?

- Size

Imagine you have some past data on quality of tomatoes. What visual features do you think will be useful?

- Size

Imagine you have some past data on quality of tomatoes. What visual features do you think will be useful?

- Size
- Colour

Imagine you have some past data on quality of tomatoes. What visual features do you think will be useful?

- Size
- Colour

Imagine you have some past data on quality of tomatoes. What visual features do you think will be useful?

- Size
- Colour
- Texture

Sample Dataset

Here is our example dataset with tomato features:

Sample	Colour	Size	Texture	Condition
1	Orange	Small	Smooth	Good
2	Red	Small	Rough	Good
3	Orange	Medium	Smooth	Bad
4	Yellow	Large	Smooth	Bad

Useful Features

Is the sample number a useful feature for predicting quality of a tomato?

Useful Features

Is the sample number a useful feature for predicting quality of a tomato?

Answer: Usually no! Sample numbers are typically arbitrary identifiers and not meaningful features. Let us remove it.

Useful Features

Is the sample number a useful feature for predicting quality of a tomato?

Answer: Usually no! Sample numbers are typically arbitrary identifiers and not meaningful features. Let us remove it.

Let us modify our data table for now.

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

The training set consists of two parts:

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

The training set consists of two parts:

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

The training set consists of two parts:

1. Features (Input Variables)

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

The training set consists of two parts:

1. Features (Input Variables)

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

The training set consists of two parts:

1. Features (Input Variables)
2. Output or Response Variable

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

We call this matrix as \mathcal{D} , containing:

1. Feature matrix ($\mathbf{X} \in \mathbb{R}^{n \times d}$) containing data of n samples each of which is d dimensional.

Training Set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad

We call this matrix as \mathcal{D} , containing:

1. Feature matrix ($\mathbf{X} \in \mathbb{R}^{n \times d}$) containing data of n samples each of which is d dimensional.
2. Output vector ($\mathbf{y} \in \mathbb{R}^n$) containing output variable for n samples.

- Feature matrix: $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}$ where $\mathbf{x}_i \in \mathbb{R}^d$

- Feature matrix: $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}$ where $\mathbf{x}_i \in \mathbb{R}^d$

- Feature matrix: $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}$ where $\mathbf{x}_i \in \mathbb{R}^d$

Data Matrix Details

- Feature matrix: $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}$ where $\mathbf{x}_i \in \mathbb{R}^d$
- Example (after encoding): $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ (Orange=1, Small=0, Smooth=1)

Data Matrix Details

- Feature matrix: $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}$ where $\mathbf{x}_i \in \mathbb{R}^d$
- Example (after encoding): $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ (Orange=1, Small=0, Smooth=1)

Data Matrix Details

- Feature matrix: $\mathbf{X} = \begin{bmatrix} \mathbf{x}_1^\top \\ \mathbf{x}_2^\top \\ \vdots \\ \mathbf{x}_n^\top \end{bmatrix}$ where $\mathbf{x}_i \in \mathbb{R}^d$
- Example (after encoding): $\mathbf{x}_1 = \begin{bmatrix} 1 \\ 0 \\ 1 \end{bmatrix}$ (Orange=1, Small=0, Smooth=1)
- Complete dataset: $\mathcal{D} = \{(\mathbf{x}_i^\top, y_i)\}_{i=1}^n$

Prediction Task

Estimate condition for unseen tomatoes (#5, 6) based on data set.

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad
Red	Large	Rough	?
Orange	Large	Rough	?

Testing Set

Testing set is similar to training set, but, does not contain labels for output variable.

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad
Red	Large	Rough	?
Orange	Large	Rough	?

Prediction Task

We hope to:

Prediction Task

We hope to:

Prediction Task

We hope to:

1. Learn f : $\text{Condition} = f(\text{colour, size, texture})$

Prediction Task

We hope to:

1. Learn f : $\text{Condition} = f(\text{colour, size, texture})$

Prediction Task

We hope to:

1. Learn f : $\text{Condition} = f(\text{colour, size, texture})$
2. From Training Dataset

Prediction Task

We hope to:

1. Learn f : $\text{Condition} = f(\text{colour, size, texture})$
2. From Training Dataset

Prediction Task

We hope to:

1. Learn f : Condition = f (colour, size, texture)
2. From Training Dataset
3. To Predict the condition for the Testing set

Colour	Size	Texture	Condition
Orange	Small	Smooth	Good
Red	Small	Rough	Good
Orange	Medium	Smooth	Bad
Yellow	Large	Smooth	Bad
Red	Large	Rough	?
Orange	Large	Rough	?

- Q: Is predicting on test set enough to say our model generalises?

- Q: Is predicting on test set enough to say our model generalises?

- Q: Is predicting on test set enough to say our model generalises?

- Q: Is predicting on test set enough to say our model generalises?
- A: Ideally, no!

- Q: Is predicting on test set enough to say our model generalises?
- A: Ideally, no!

- Q: Is predicting on test set enough to say our model generalises?
- A: Ideally, no!
- Ideally - we want to predict “well” on all possible inputs. But, can we test that?

- Q: Is predicting on test set enough to say our model generalises?
- A: Ideally, no!
- Ideally - we want to predict “well” on all possible inputs. But, can we test that?

- Q: Is predicting on test set enough to say our model generalises?
- A: Ideally, no!
- Ideally - we want to predict “well” on all possible inputs. But, can we test that?
- No! Since, the test set is only a sample from all possible inputs.

Generalisation

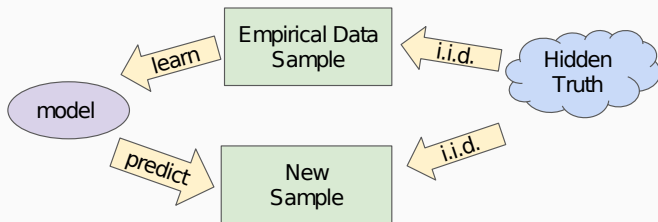


Image courtesy Google ML crash course

Generalisation

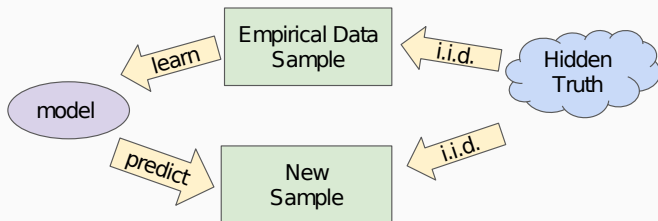


Image courtesy Google ML crash course

Both the training set and the test set are samples drawn from the hidden true distribution (also sometimes called population)

Generalisation

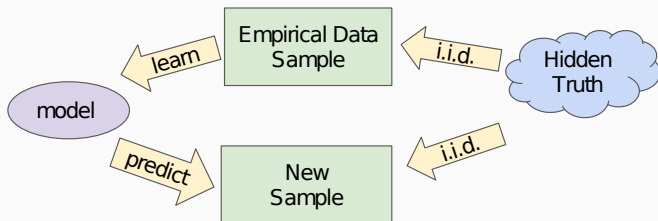


Image courtesy Google ML crash course

Both the training set and the test set are samples drawn from the hidden true distribution (also sometimes called population)

More discussion later once we study bias and variance

Second ML Task: Predict energy consumption of campus

Question: What factors does the campus energy consumption depend on?

Answer:

Second ML Task: Predict energy consumption of campus

Question: What factors does the campus energy consumption depend on?

Answer:

Second ML Task: Predict energy consumption of campus

Question: What factors does the campus energy consumption depend on?

Answer:

- # People (More people \implies More Energy)

Second ML Task: Predict energy consumption of campus

Question: What factors does the campus energy consumption depend on?

Answer:

- # People (More people \implies More Energy)

Second ML Task: Predict energy consumption of campus

Question: What factors does the campus energy consumption depend on?

Answer:

- # People (More people \implies More Energy)
- Temperature (Higher Temp. \implies Higher Energy)

Second ML Task: Predict energy consumption of campus

Question: What factors does the campus energy consumption depend on?

Answer:

- # People (More people \implies More Energy)
- Temperature (Higher Temp. \implies Higher Energy)

# People	Temp (C)	Energy (kWh)
4000	30	30
4200	30	32
4200	35	40
3000	20	?
1000	45	?

Classification vs Regression

- Classification

Classification vs Regression

- Classification

Classification vs Regression

- Classification

Classification vs Regression

- Classification
 - Output variable is discrete

Classification vs Regression

- Classification
 - Output variable is discrete

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous
 - i.e. $y_i \in \mathbb{R}$

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous
 - i.e. $y_i \in \mathbb{R}$

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous
 - i.e. $y_i \in \mathbb{R}$
 - Examples - Predicting:

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous
 - i.e. $y_i \in \mathbb{R}$
 - Examples - Predicting:

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous
 - i.e. $y_i \in \mathbb{R}$
 - Examples - Predicting:
 - How much energy will campus consume?

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous
 - i.e. $y_i \in \mathbb{R}$
 - Examples - Predicting:
 - How much energy will campus consume?

Classification vs Regression

- Classification
 - Output variable is discrete
 - i.e. $y_i \in \{1, 2, \dots, k\}$ where k is number of classes
 - Examples - Predicting:
 - Will I get a loan? (Yes, No)
 - What is the quality of fruit? (Good, Bad)
- Regression
 - Output variable is continuous
 - i.e. $y_i \in \mathbb{R}$
 - Examples - Predicting:
 - How much energy will campus consume?
 - How much rainfall will fall?

Metrics for Classification

Prediction (\hat{y})	Ground Truth (y)
Good	Good
Good	Good
Good	Bad
Good	Bad
Bad	Bad

Ground Truth: From the actual training set

Prediction: Made by the model

Accuracy

Prediction (\hat{y})

✓	Good
✓	Good
	Good
	Good
✓	Bad

Ground Truth (\mathbf{y})

Good
Good
Bad
Bad
Bad

Accuracy

	Prediction (\hat{y})	Ground Truth (y)
✓	Good	Good
✓	Good	Good
	Good	Bad
	Good	Bad
✓	Bad	Bad

$$\begin{aligned}\text{Accuracy} &= \frac{|\{i : y_i = \hat{y}_i\}|}{n} \\ &= \frac{3}{5} = 0.6\end{aligned}$$

Mathematical Notation: Set Cardinality and Indicator Functions

Mathematical Notation: Set Cardinality and Indicator Functions

Mathematical Notation: Set Cardinality and Indicator Functions

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
 - Reads as: “Number of indices i such that $y_i = \hat{y}_i$ ”

Mathematical Notation: Set Cardinality and Indicator Functions

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
 - Reads as: “Number of indices i such that $y_i = \hat{y}_i$ ”
 - Counts how many samples satisfy the condition

Mathematical Notation: Set Cardinality and Indicator Functions

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
 - Reads as: “Number of indices i such that $y_i = \hat{y}_i$ ”
 - Counts how many samples satisfy the condition

Mathematical Notation: Set Cardinality and Indicator Functions

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
 - Reads as: “Number of indices i such that $y_i = \hat{y}_i$ ”
 - Counts how many samples satisfy the condition

Mathematical Notation: Set Cardinality and Indicator Functions

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
 - Reads as: “Number of indices i such that $y_i = \hat{y}_i$ ”
 - Counts how many samples satisfy the condition
- **Alternative: Indicator function notation**

$$\text{Accuracy} = \frac{\sum_{i=1}^n \mathbf{1}[y_i = \hat{y}_i]}{n}$$

$$\text{where } \mathbf{1}[\text{condition}] = \begin{cases} 1 & \text{if condition is true} \\ 0 & \text{if condition is false} \end{cases}$$

Mathematical Notation: Set Cardinality and Indicator Functions

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
 - Reads as: “Number of indices i such that $y_i = \hat{y}_i$ ”
 - Counts how many samples satisfy the condition
- **Alternative: Indicator function notation**

$$\text{Accuracy} = \frac{\sum_{i=1}^n \mathbf{1}[y_i = \hat{y}_i]}{n}$$

$$\text{where } \mathbf{1}[\text{condition}] = \begin{cases} 1 & \text{if condition is true} \\ 0 & \text{if condition is false} \end{cases}$$

Mathematical Notation: Set Cardinality and Indicator Functions

- **Set cardinality notation:** $|\{i : y_i = \hat{y}_i\}|$
 - Reads as: “Number of indices i such that $y_i = \hat{y}_i$ ”
 - Counts how many samples satisfy the condition
- **Alternative: Indicator function notation**

$$\text{Accuracy} = \frac{\sum_{i=1}^n \mathbf{1}[y_i = \hat{y}_i]}{n}$$

$$\text{where } \mathbf{1}[\text{condition}] = \begin{cases} 1 & \text{if condition is true} \\ 0 & \text{if condition is false} \end{cases}$$

- Both notations are mathematically equivalent and commonly used in ML literature

Types of Data: Imbalanced Classes

1 sample {
100 samples {
Bad
Good
Good
...
Good

Imbalanced Classes

Types of Data: Imbalanced Classes

1 sample {
100 samples {
Bad
Good
Good
...
Good

Imbalanced Classes

Cases for this:

- Cancer Screening

Types of Data: Imbalanced Classes

1 sample {
100 samples {
Bad
Good
Good
...
Good

Imbalanced Classes

Cases for this:

- Cancer Screening
- Planet Detection

Accuracy Metrics: Precision

	Prediction (\hat{y})	Ground Truth (y)
→ ✓	Good	Good
→ ✓	Good	Good
→	Good	Bad
→	Good	Bad
	Bad	Good

$$\text{Precision} = \frac{|\{i : y_i = \hat{y}_i = \text{Good}\}|}{|\{i : \hat{y}_i = \text{Good}\}|} = \frac{2}{4} = 0.5$$

“the fraction of relevant instances among the retrieved instances”,
i.e. “out of the number of times we predict Good, how many times
is the condition actually Good”

Accuracy Metrics: Precision

	Prediction (\hat{y})	Ground Truth (y)
→ ✓	Good	Good
→ ✓	Good	Good
→	Good	Bad
→	Good	Bad
	Bad	Good

$$\text{Precision} = \frac{|\{i : y_i = \hat{y}_i = \text{Good}\}|}{|\{i : \hat{y}_i = \text{Good}\}|} = \frac{2}{4} = 0.5$$

“the fraction of relevant instances among the retrieved instances”,
i.e. “out of the number of times we predict Good, how many times
is the condition actually Good”

Accuracy Metrics: Recall

	Prediction (\hat{y})	Ground Truth (y)
→ ✓	Good	Good
→ ✓	Good	Good
	Good	Bad
	Good	Bad
→	Bad	Good

$$\text{Recall} = \frac{|\{i : y_i = \hat{y}_i = \text{Good}\}|}{|\{i : y_i = \text{Good}\}|} = \frac{2}{3} = 0.67$$

“the fraction of the total amount of relevant instances that were actually retrieved”

Types of Data: Imbalanced Classes

Given predictions of whether a tissue is cancerous or not ($n = 100$).

$$\begin{array}{cc} \text{Prediction } (\hat{y}) & \text{Ground Truth } (\mathbf{y}) \\ \rightarrow \left(\begin{array}{c} \text{Yes} \\ \text{No} \\ \text{No} \\ \dots \\ \text{No} \end{array} \right) & \rightarrow \left(\begin{array}{c} \text{No} \\ \text{No} \\ \dots \\ \text{No} \\ \text{Yes} \end{array} \right) \end{array}$$

Types of Data: Imbalanced Classes

Given predictions of whether a tissue is cancerous or not ($n = 100$).

Prediction (\hat{y})	Ground Truth (\mathbf{y})
→ $\begin{pmatrix} \text{Yes} \\ \text{No} \\ \text{No} \\ \dots \\ \text{No} \end{pmatrix}$	$\begin{pmatrix} \text{No} \\ \text{No} \\ \dots \\ \text{No} \\ \text{Yes} \end{pmatrix}$

$$\text{Accuracy} = \frac{98}{100} = 0.98$$

$$\text{Recall} = \frac{0}{1} = 0$$

$$\text{Precision} = \frac{0}{1} = 0$$

Accuracy Metrics: Confusion Matrix

		Ground Truth	
		Yes	No
Predicted	Yes	0	1
	No	1	98

Accuracy Metrics: Confusion Matrix

		Ground Truth	
		Yes	No
Predicted	Yes	0	1
	No	1	98

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

Accuracy Metric: Confusion Matrix

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

$$\text{Precision} = \frac{TP}{TP+FP}$$

Accuracy Metric: Confusion Matrix

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

$$\text{Precision} = \frac{TP}{TP+FP}$$

Accuracy Metric: Confusion Matrix

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

$$\text{Recall} = \frac{TP}{TP+FN}$$

Accuracy Metric: Confusion Matrix

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

$$\text{Recall} = \frac{TP}{TP+FN}$$

Accuracy Metrics: F-Score

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

$$F\text{-Score} = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Accuracy Metrics: Matthew's Correlation Coefficient

		Ground Truth	
		Yes	No
Predicted	Yes	True Positive	False Positive
	No	False Negative	True Negative

Matthew's correlation coefficient =

$$\frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}}$$

Accuracy Metrics: Example

For the data given below, calculate:

	G.T. Positive	G.T. Negative
Pred Positive	90	4
Pred Negative	1	1

Precision = ?

Recall = ?

F-Score = ?

Matthew's Coeff. = ?

Accuracy Metrics: Answer

For the same data

	G.T. Positive	G.T. Negative
Pred Positive	90	4
Pred Negative	1	1

$$\text{Precision} = \frac{90}{94}$$

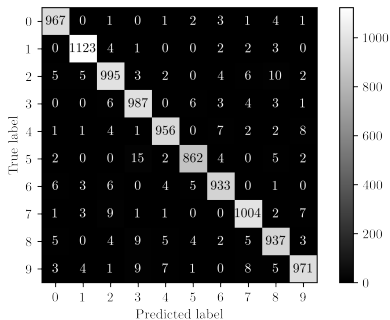
$$\text{Recall} = \frac{90}{91}$$

$$\text{F-Score} = 0.9524$$

$$\text{Matthew's Coeff.} = 0.14$$

Confusion Matrix for multi-class classification

Notebook: [confusion-mnist.html](#)



Metrics for Regression MSE & MAE

Prediction (\hat{y})	Ground Truth (y)
10	20
20	30
30	40
40	50
50	60

$$\text{Mean Squared Error (MSE)} = \frac{\sum_{i=1}^n (\hat{y}_i - y_i)^2}{n}$$

$$\text{Root Mean Square Error (RMSE)} = \sqrt{\text{MSE}}$$

Accuracy Metrics: MAE & ME

Prediction (\hat{y})
10
20
30
40
50

Ground Truth
20
30
40
50
60

$$\text{Mean Absolute Error (MAE)} = \frac{\sum_{i=1}^n |\hat{y}_i - y_i|}{n}$$

$$\text{Mean Error} = \frac{\sum_{i=1}^n (\hat{y}_i - y_i)}{n}$$

Accuracy Metrics: MAE & ME

Prediction (\hat{y})	Ground Truth
10	20
20	30
30	40
40	50
50	60

$$\text{Mean Absolute Error (MAE)} = \frac{\sum_{i=1}^n |\hat{y}_i - y_i|}{n}$$

$$\text{Mean Error} = \frac{\sum_{i=1}^n (\hat{y}_i - y_i)}{n}$$

Is there any downside with using mean error?

Accuracy Metrics: MAE & ME

Prediction (\hat{y})
10
20
30
40
50

Ground Truth
20
30
40
50
60

$$\text{Mean Absolute Error (MAE)} = \frac{\sum_{i=1}^n |\hat{y}_i - y_i|}{n}$$

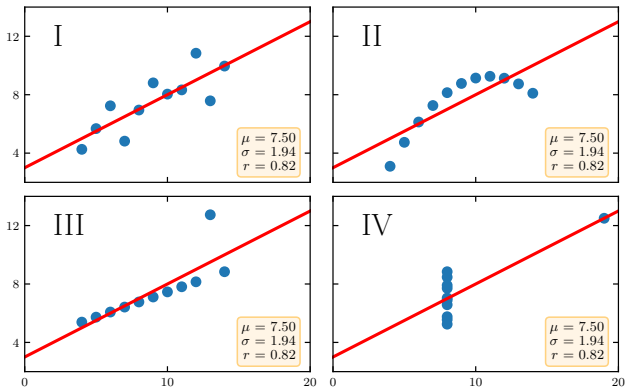
$$\text{Mean Error} = \frac{\sum_{i=1}^n (\hat{y}_i - y_i)}{n}$$

Is there any downside with using mean error?

Errors can get cancelled out

The Importance of Plotting

Notebook: [anscombe.html](#)



Anscombe's Quartet

Notebook: dummy-baselines.html

The Importance of Plotting

Property	Value	Across datasets
mean(X)	9	exact
mean(Y)	7.5	up to 3 decimal places
Linear regression line	$y = 3.00 + 0.500x$	up to 2 decimal places